A Classification-based Framework for Learning Object Assembly

Roderick A. Farmer and Baden Hughes
Department of Computer Science and Software Engineering
The University of Melbourne
Parkville, VIC 3010, Australia
{raf, badenh}@cs.mu.oz.au

Abstract

Relations between learning outcomes and the learning objects which are assembled to facilitate their achievement are the subject of increasingly prevalent investigation, particularly with approaches which advocate the aggregation of learning objects as complex constituencies for achieving learning outcomes. From the perspective of situated learning, we show how the CASE framework imbues learning objects with a closed set of properties which can be classified and aggregated into learning object assemblies in a principled fashion. We argue that the computational and pedagogical tractability of this model provides a new insight into learning object evaluation, and hence learning outcomes.

1. Introduction

Motivated by the apparent absence of instructional design and learning theories in learning object research [7], we propose that the ability to construct aggregates of learning objects is greatly restricted by our present inability to describe objects in terms of their contextual or situated use. Our work considers the need for a closed set of properties that can be used to combine learning objects into complex assemblies. Here we propose a framework that allows the evaluation of these aggregates in terms of qualitatively discrete learning outcomes.

To model these properties, we use a situated task analysis model, CASE [2], which provides an integrative framework for modelling learning object properties against situated learning constraints. We discuss the constitution of learning object assemblies and present an algebraically-based framework for formally constructing learning object assemblies using CASE properties.

Support for relationships between design and use is often lacking in many computer-supported learning environments and is described by [1] as the "social-technical" gap. Learning objects possess the ability to bridge this gap through the qualitative description of object properties that meet the requirements of both domains. These properties can be sourced from a number of specifications and standards, including IMS [5] and LOM [4].

However, we can assume that if these properties do not assist us in describing learning objects in terms of discrete learning outcomes, then the process of learning object selection, sequencing and association is likely to be problematic. We propose therefore, that frameworks which assist in this process represent a contribution to the field. One such framework described in [2] is CASE, a situated task analysis framework for analysing learning-computer interaction in computer-mediated learning environments. CASE represents a situated task analysis framework for describing learning objects and their subsequent evaluation. Its strength comes from its ability to describe learning objects in terms of domain-specific learning outcomes, as well as its consideration of situated learning theory in learning object design and implementation.

CASE proposes that there are four critical qualitative components to learning object design and evaluation: Cognition, Activity, Social Organisation and Environment. Here, each component depicts a level of investigation into the socio-technical constraints which are likely to impact learning object design in meeting various learning outcomes. Elemental characterisitcs of these components, and their application within a whole-of-lifetime task analysis methodology are described in [2].

2. Learning Object Assembly

Aggregation of learning objects requires consideration of object selection, sequencing and association [3]. Additionally, these processes must match a set of criteria which describe discrete learning outcomes. Existing research has examined how such assemblies can be generated from general, educational and classification metadata [3], yet currently lacks discussion of frameworks for describing situational constraints likely to impact learning object reuse.

Describing learning object qualities in terms of specific properties affords the establishment of formal relationships. These relationships can be used to construct assemblies of varying complexity and constituency. Given a learning object with the CASE attribute types Cognition, Activity, Social Organisation, and Environment, we observe $Obj(C,A,S,E)$. Each CASE type in turn may represent a set of related properties p_k, such that $Obj(C(p_k),A(p_k),S(p_k),E(p_k))$. Consequently, a simple view of multi-component learning object assemblies (A) may then be seen as having the aggregate properties of all objects:

$$A \equiv \sum_{i=1}^{n} Obj_i(C(p_k),A(p_k),S(p_k),E(p_k)) \qquad (1)$$

Equation 1 provides a means of describing large sets of learning objects in terms of their proposed learning function, and according to a number of a priori conditions during selection, sequencing or association.

Learning object assembly can be performed in a number of different ways, depending on the type of task. Some proto-typical tasks that can be performed based on the formalism presented above include 1) Finding assemblies of learning objects which meet a defined learning outcome; 2) Grouping learning objects by similar or different property sets; 3) Grouping assemblies of learning objects by similar or different property sets; 4) Excluding learning objects from assemblies based on properties; 5) Deriving ranked lists of objects meeting certain requirements based on properties; and 7) Performing preferential object selection based on a property-based weighting scheme.

Our approach is motivated by the need to determine not only relations between individual learning objects, but also between learning object assemblies and specified learning outcomes. As others have found it cost-ineffective for developers to manually determine relationships between learning objects [3], we propose a formal approach based upon instance-based learning – where one seeks to establish groups of objects based on some measurable degree of similarity – to dynamically classify learning objects across a set of closed properties. Importantly, we limit our discussion in this paper to that of defining and describing the process by which to formulate the similarity measure.

Our approach includes three sub-processes in the evaluation of similarity. Firstly, we wish to establish similarity between learning objects based upon the socio-technical CASE properties of learner-computer interaction. Second, we wish to evaluate both structural and semantic degrees of object similarity. As such, we are able to evaluate object similarity across the full range of object properties. These processes can be represented algebraically as follows.

We assume an arbitrary number n learning objects in a repository $\mathcal{X}$, such that $f : \mathbb{R}^n \rightarrow \mathcal{X}$ where $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$. Let x_i be an object instance where $x_i \in \mathcal{X}$.

Each instance x_i contains a feature vector $\mathcal{F}_i$ that describes the set of attribute types which represent the learning properties of the object (Equation 2):

$$\mathcal{F}_i = a_1(x_i), a_2(x_i), \ldots, a_n(x_i) \qquad (2)$$

where $a_r(x)$ is the value of the rth attribute in x_i. Each attribute type a_r maintains a set of real values we call properties $\mathcal{P}$, such that $\mathcal{P} = \{a_r.p_1, \ldots, a_r.p_k\}$. As we assume our attributes to represent real values in the feature space, we can calculate $a_r(x_i)$ using a traditional similarity measure such as Euclidean distance:

$$d(a_r(x_i), a_r(x_j)) \equiv A\sqrt{\sum_{k=1}^{n} (a_r p_k(x_i) - a_r p_k(x_j))^2} \qquad (3)$$

where A is some predetermined weight used for feature selection. If we are not interested in weighting the importance of different attributes within the feature vector, we set $A = 1$.

In determining the similarity of two objects, we need to calculate the content ($\mathcal{S}_c$), structural ($\mathcal{S}_s$), and semantic ($\mathcal{S}_m$) similarities. The content similarity $\mathcal{S}_c$ is an evaluation of the similarity between socio-technical feature vectors $x_i.\mathcal{F}_i$.

Let the similarity or *relatedness* between two instances be defined as:

$$\mathcal{R} = \alpha \mathcal{S}_c + \beta \mathcal{S}_s + \gamma \mathcal{S}_m \qquad (4)$$

where α, β, γ are constants such that $\alpha + \beta + \gamma = 1$. By normalising the weights assigned to each evaluative subprocess we are able to easily customise how we determine object relatedness. For instance, in order to aggregate learning object instances solely upon structural relationships, we simply set $\alpha = \gamma = 0$. If we wish to return only assemblies which match learning outcomes according to various socio-technical conditions of use, we set $\beta = \gamma = 0$.

Now that we have defined the overarching relatedness function (Equation 4), we proceed with providing a definition of the content similarity function $\mathcal{S}_c$:

$$\mathcal{S}_c = d(x_i, x_j) \qquad (5)$$

Here, Equation 5 states that the content similarity can be established by determining the distance d between two object instances x_i, x_j. To establish the *distance* between these instances, we proceed as in Equation 3:

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2} \qquad (6)$$

Equation 6 establishes the content similarity of two objects based on the Euclidean distance of the attribute types of the feature set. It can be seen from (3) that the distance is in fact calculated by the attribute type properties.

This approach is able to maximise the utility of the previously described prototypical tasks one may wish to perform with learning object assemblies. It permits aggregation of learning objects based on a discrete set of closed discrete properties, whether they be content, structural or semantic. We can weight the importance of property attributes according to some predetermined (or learnt) domain-specific criteria. This approach can be used to not only return a best-match learning object assembly given a desired learning outcome, but also a ranked list of assemblies. This point is significant as it is often instructive in component-based software development to know the maximum set of objects that will never meet a given criteria.

3. Evaluation

The tractability of our approach depends on several assumptions. Our approach assumes that the attributes of an object can be represented as discrete real numbers. However, many attributes do not meet this condition. This problem can be addressed by applying some transformation function $\tau(x_i)$ across the learning object space, $\mathcal{X}$, converting all qualitative values into reals. This is a standard technique used in many data clustering approaches. We also assume that each object in the learning object space has a homogeneous set of attribute types and properties. This is important as it implies a constant attribute vector length when comparing objects and greatly simplifies calculating learning object similarities.

There are a range of benefits to our approach. Unlike existing approaches [3, 6], our formalism for aggregation of learning objects is not dependent upon other existing subsystems. Furthermore, these aggregations provide the necessary association required to bridge system design, instructional and learning theory, providing a critical extension to previous work in the area [7]. Furthermore, our approach reduces the ambiguity associated with component-reuse in learning environments by providing educators with an effective means of evaluating the utility of a set of learning objects against desired learning outcomes. Our approach satisfies a primary goal of evaluation by providing the ability to apply qualitative heuristics to learning objects independent of implementation and at varying degrees of abstraction.

4. Conclusion and Future Work

We have examined the need for a closed set of properties which can be applied to describe, classify and manipulate learning object assembliess. The proposed formal framework we have presented represents a major step towards improved development, evaluation and personalisation of learning object assemblies.

Further work is still required to transform the abstract properties of the CASE model into discrete attribute-value pairs that can be incorporated into existing metadata schemes. Developer tools such as activity checklists and qualitatively-motivated templates for learning (multi-tier) object assemblies will be central to this process. Additionally, learners are usually inclined to subjectively assume that some information is more authoritative than other sources. This principal can also be applied to learning object assemblies. Although our approach currently groups like objects into collections, highly-related yet dissimilar objects would not be grouped. Existing machine learning approaches such as link analysis may represent a means of treating this problem.

The assignment of weights to each of the sub-processes in our framework affords system learnability whereby collaborative filtering, user preferences and historical actions can be used to train the system to return a more tailored set of results. This issue will be addressed in conjunction with our research into system optimality and efficiency. Finally, and arguably the most complex issue facing end user developers is that of establishing meaningful sequences of learning objects for some learning activity. Providing means of automating this process represents a major challenge.

References

[1] M. S. Ackerman. The intellectual challenge of cscw: The gap between social requirements and technical feasibility. *Human-Computer Interaction*, 15(2-3):179–203, 2000.

[2] R. A. Farmer. Situated task analysis in learner-centred call. In P. Zaphiris, editor, *User-Centred Computer Aided Language Learning*. Idea Group Publishing, Hershey, PA, 2006. In Press.

[3] R. Farrell, S. D. Liburd, and J. C. Thomas. Dynamic assembly of learning objects. In *Proceedings of World Wide Web Conference 2004 (WWW2004)*, New York, NY, 2004. AACE Digital Library.

[4] IEEE. Draft standard for learning object metadata - proposed standard. Technical report, IEEE, Piscataway, NJ, 2002.

[5] IMS Global Learning Consortium. *IMS Meta-data Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata. Version 1.3 Public Draft*, May 2004. http://www.imsglobal.org/metadata/. Last accessed January 30th 2005.

[6] S. Sánchez and M.-Á. Sicilia. On the semantics of aggregation and generalization in learning object contracts. In *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'04)*, pages 425–429. IEEE Computer Society Press, 2004.

[7] D. A. Wiley. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley, editor, *Instructional use of Learning Objects*. Agency for Instructional Design, Bloomington, US, 2002.